

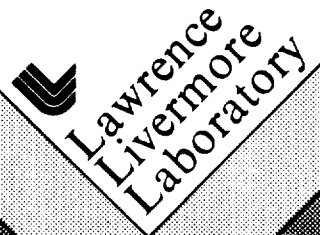
CIRCULATION COPY
SUBJECT TO RECALL
IN TWO WEEKS

UCID-18549

Analysis of a 2-D Code on the CRAY-1

Tim Rudy

February 19, 1980

The logo for Lawrence Livermore Laboratory, featuring a stylized 'L' symbol and the text 'Lawrence Livermore Laboratory' arranged in a triangular shape.

Lawrence
Livermore
Laboratory

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the Laboratory.

Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore Laboratory under Contract W-7405-Eng-48.

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This report has been reproduced
directly from the best available copy.

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information
P.O. Box 62, Oak Ridge, TN 37831
Prices available from (615) 576-8401, FTS 626-8401

Available to the public from the
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd.,
Springfield, VA 22161

ANALYSIS OF A 2-D CODE ON THE CRAY-1

ABSTRACT

This paper describes the results of the conversion of a 2-D time dependent finite difference code from a CDC 7600 to the CRAY-1. The performance improvement, degree of vectorization and output characteristics on the CRAY-1 are presented.

INTRODUCTION

The recent acquisition of a CRAY-1 by the Livermore Computer Center offered the user community the potential for a significant improvement in thruput capabilities for application codes developed for the CDC 7600.

This paper describes the performance enhancement, degree of vectorization and output characteristics of a 2-D time dependent finite difference code on the CRAY-1. Measurements were made of instruction type, vector length, megaflop rate and output rates for various forms of output on the CRAY-1.

The programs can be viewed as three types of calculational phases. Phase 1 is completely explicit. The second phase is composed of explicit and implicit equations. In both Phase 1 and 2 a high degree of vectorization was attained. The third phase is composed of scalar, short vector and long vector operations.

Comparisons to the 7600 version are somewhat misleading since the 7600 code was completely coded in assembly language. The CRAY-1 version is written in FORTRAN and uses the CFT [1] compiler. Approximately 300 lines of assembly code is used on the CRAY-1 for performance improvements to small kernels in the program.

INSTRUCTION MIX

In Table 1 the instruction mix for a "typical" problem is described. The problem consisted of 6 energy groups, which dominate Phase 3 calculations, and a physical grid which is 19 x 63 grid points.

Table 1: Instruction Mix

	Phase 1	Phase 2 (percent)	Phase 3
<u>Distribution of vector operations</u>			
Short vector VL=6,7			85
Medium vector VL=18,19		39	1
Long vector VL=62,63	100	61	14
<u>Distribution of floating point operations</u>			
Scalar + scalar	1		7
Scalar - scalar	2	1	1
Scalar x scalar			8
1/scalar		1	
Scalar/vector	4	6	
Scalar x vector	10	8	35
Vector x vector	33	44	14
Vector + vector	25	24	30
Vector - vector	21	3	
Scalar + vector		6	4
Scalar - vector	4	7	1

The instruction mix was obtained by enabling the "W" option in the CFT compiler. This option generates calls to external routines for the various floating point operations tabulated in Table 1. An assembly language routine was coded to perform the arithmetic operations and count the number of calls for each floating point operation.

Although Phase 1 and 2 calculations are highly vectorized the presence of indirect addressing and data dependencies in Phase 3 calculations reduced

the degree of vectorization to 85%. Alternate methods are being investigated to remedy this situation.

Another observation is that the distribution of vector instruction type is significantly different for Phase 3 calculations versus Phase 1 and 2.

MEGAFLOP RATE

In Table 2 the megaflop rate for the different phases of the code are described. Floating point divides, square root, exponential and log are each counted as four floating point operations. Since the code typically subcycles Phase 3 calculations we normally see five Phase 1 and 2 cycles to every Phase 3 pass. This implies that our average megaflop rate is approximately 11.3.

Table 2: Megaflops

	Phase 1	Phase 2	Phase 3
Primitive flops	1.47 E+6	1.45 E+6	9.16 E+6
Square root, exp, log flops	.11 E+6	.06 E+6	.07 E+6
Total flops	1.58 E+6	1.51 E+6	9.23 E+6
Compute time (MSEC)	81.4	95.5	1296
Megaflops	19.4	15.8	7.12

ASSEMBLY OPTIMIZATION

Approximately 300 lines of assembly language was written to enhance the thruput of the CRAY version of the code. The three primary routines written in assembly language are:

1. EOSTLU - gather coefficients for solving the biquadratic equation in Equation of State calculations.
2. INV - invert a 6 x 6 or 10 x 10 matrix by Gauss elimination.
3. SIG - performs a dot product on each zone to collapse an array from 5 dimensions (i,j,k,R,Z) to four dimensions (j,k,R,Z). This is a short vector operation.

EOSTLU is used primarily in Phase 1 and 2 calculations. INV and SIG dominate the compute time in Phase 3.

The time required to invert 6 x 6 and 10 x 10 matrices on the 7600 and CRAY is presented in Table 3.

Table 3: Matrix Inverse (USEC)

Size of Matrix	6 x 6	10 x 10
7600 Fortran	379	1483
7600 Assembly	120	507
CRAY Fortran	108	298
CRAY Assembly	31	99

The SIG algorithm is approximately 2.7 times faster in assembly than vectorized CRAY FORTRAN and 10 to 12 times CRAY FORTRAN scalar.

With the assembly INV and SIG routines 24% of the Phase 3 execution time is consumed by these two routines. If the vectorized FORTRAN versions of these routines were used the execution time for Phase 3 would increase from 1.296 seconds per time step to 1.992 seconds.

All assembly routines have FORTRAN equivalents. By setting a parameter at compile time programmers may use either version.

CODE OUTPUT

The output generated by this code, controlled by the user, is composed of:

1. Alphanumeric - printed output describing the state of the problem in global, regional and local (e.g., zone) detail.
2. Graphics - grid, velocity vector and isoplots are generated for the full grid and/or subsets of the grid. At problem termination a set of time history plots are also provided.
3. Binary - binary files are written to allow the user to restart the problem, post process the state of the problem or link to another code. Historically, restart dump frequency was chosen as a function of machine reliability.

For a "typical" problem the following measurements were made for a completed problem:

Table 4: Code Output

Output Type	Bits of Output	Kilobits/second (KBS)
Alphanumeric	26.5 E+6	16.1
Graphics	62.1 E+6	37.7
Binary	80.4 E+6	48.8
Total	169.0 E+6	102.6

Since this particular code rarely links to another code the frequency of binary dumps was removed from user control and replaced by the elapsed time used since the last dump. Under user control binary dumps were being written every 1.5 to 5 minutes on the CRAY.

To permit post processing of a particular calculation the restart dumps, selected by the user, were replaced by binary files which contained a reduced set of output. These files are typically four times smaller than complete restart dumps.

This change in output generation reduced the binary data output from 80.4 megabits to 19.6 megabits. The output rate was similarly reduced from 102.6 KBS to 65.7 KBS.

The time used in alphanumeric routines was 1% of the total time used. Graphics output required approximately 6% of the total execution time. Each frame of graphics output, on average, required 113000 bits.

If we extrapolate to a future high speed computer which is an order of magnitude faster than a CRAY-1, the output rate would be 657 KBS. If this output rate is "typical" for application codes on such a machine a terabit storage device would be filled in less than 18 days; assuming all output is stored on a device. Clearly, new methods should be investigated in the presentation of calculational results to users.

REFERENCES

- [1] CRAY-1 Computer System, reference manual, 2240009, CRAY Research Inc., 1978.